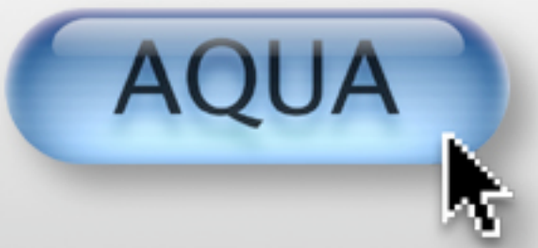
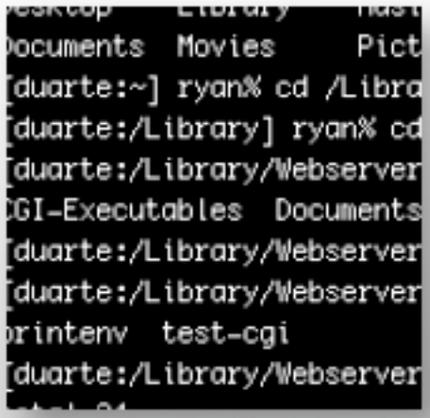
Cocoa Update





Cocoa Update

Ali Ozer
Manager
Cocoa Frameworks

What You'll See

- What's new in Cocoa in Panther
  - AppKit
  - Foundation
- What will be covered
  - New features
  - Significant API changes
  - But not bug fixes
- Will provide pointers to other sessions

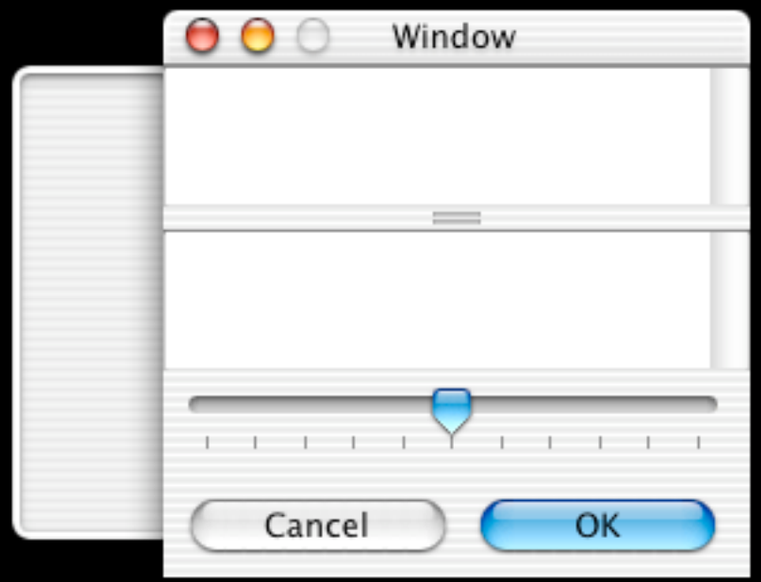
AppKit Update

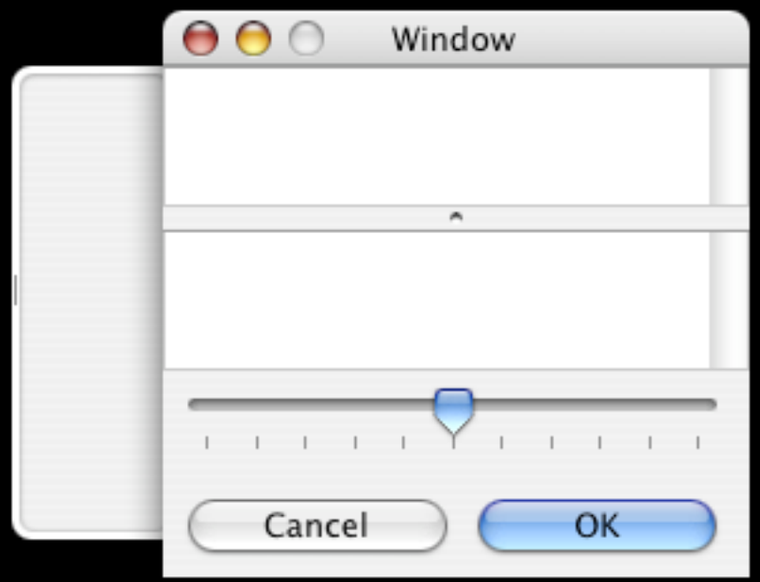
Ali Ozer
Manager
Cocoa Frameworks

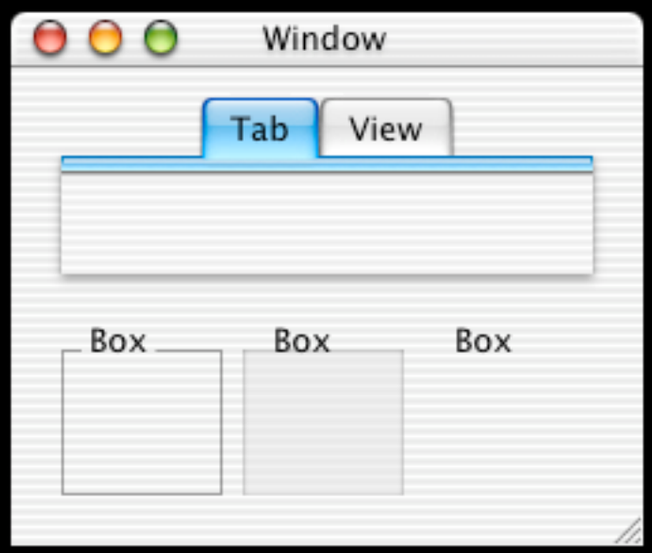
Aqua Changes

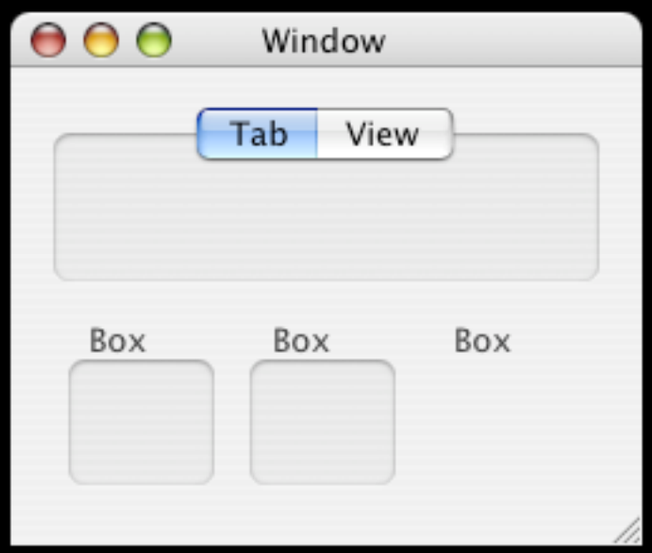
- Refining the look of Aqua
- Changes in the looks of
  - Window, menu
  - Tab view
  - Box
  - Push button, slider, splitter, drawer
- Automatic adoption by applications
  - Compatible metrics
  - Compatible APIs

Aqua Changes







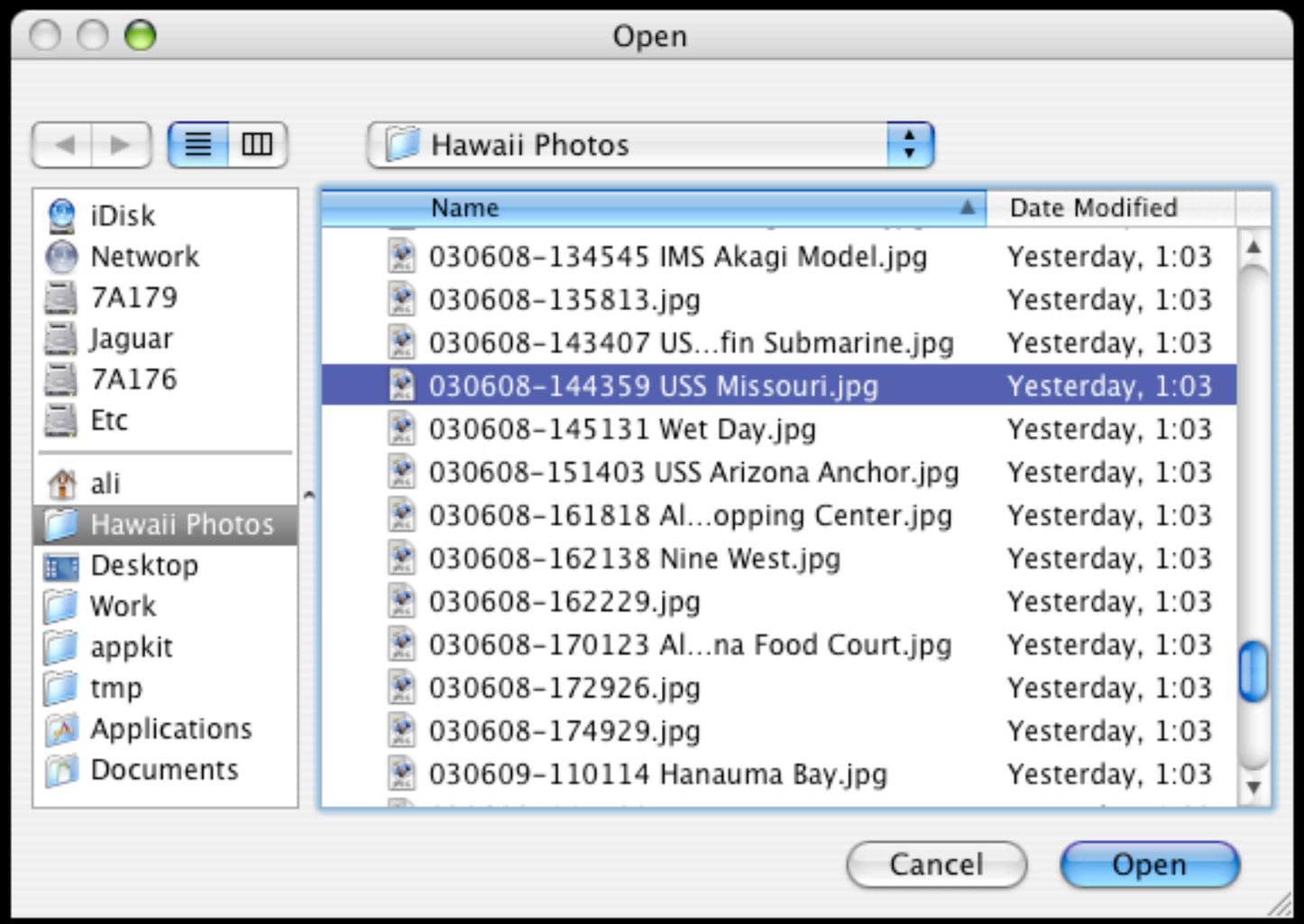


Aqua Changes

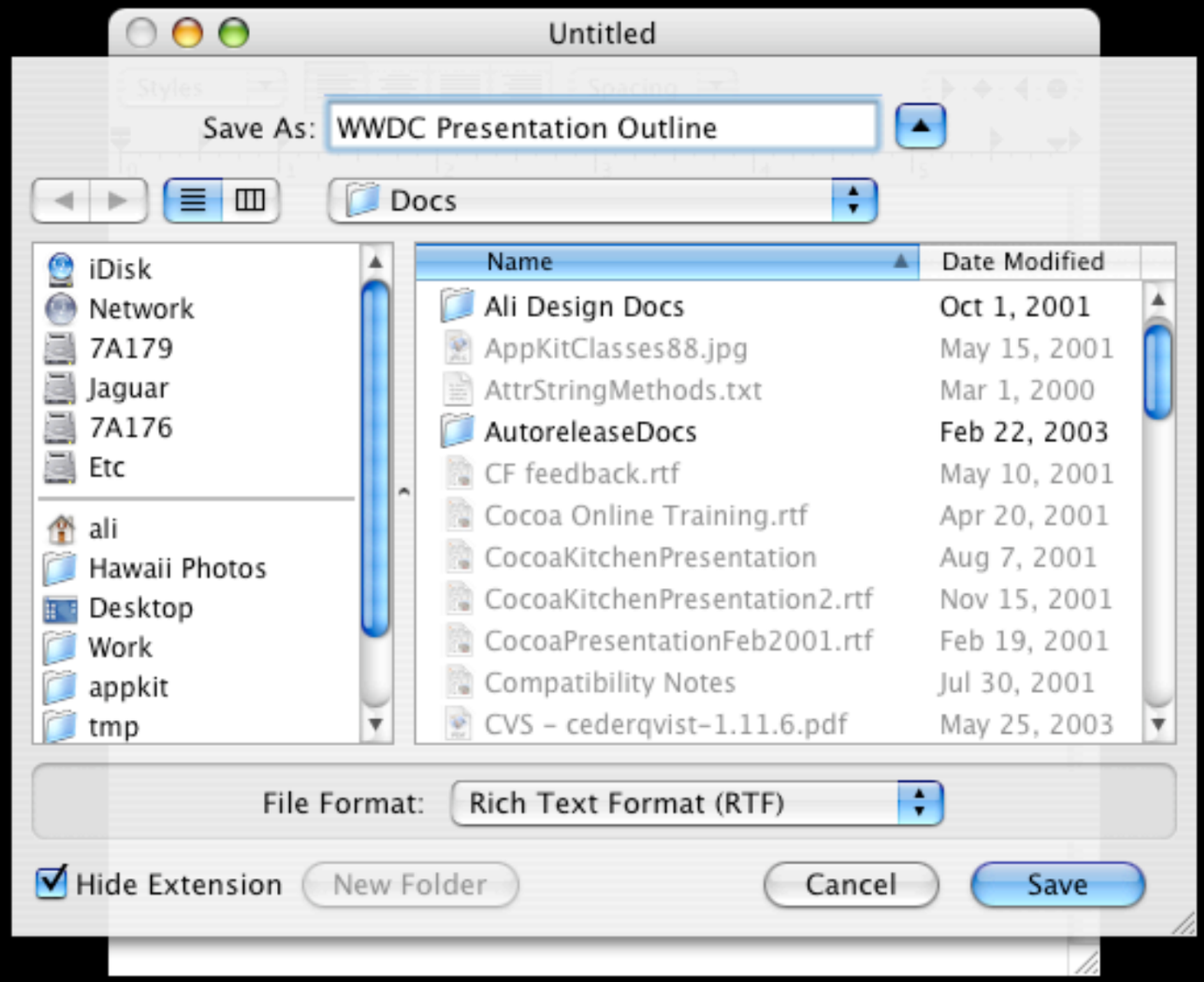
- Smooth scrolling
- Faster window and sheet animations
- Distinct look for “main”, “key”, and “inactive” windows

NSSavePanel, NSOpenPanel

- Brand new look and feel, compatible APIs



NSSavePanel, NSOpenPanel



NSSavePanel, NSOpenPanel

- New API to allow creating new folders in open panel

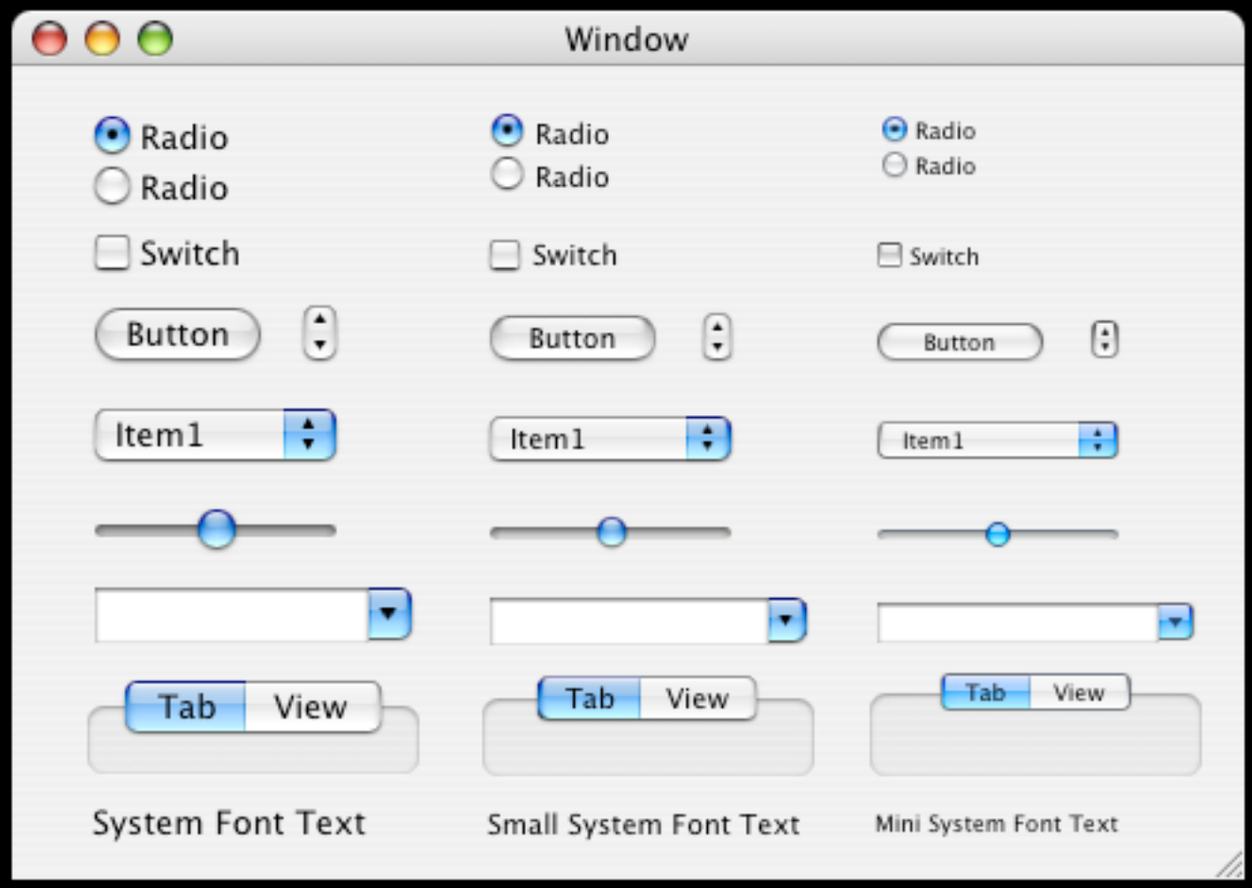
```
- (void)setCanCreateDirectories:(BOOL)flag;
- (BOOL)canCreateDirectories;
```

- Deprecated—view tags for accessing UI elements

```
NSFileHandlingPanelImageButton
NSFileHandlingDiskEjectButton
...
```

Mini Controls

- Smaller than small
  - Radio button
  - Check box
  - Push button
  - Slider
  - Tab view
  - Stepper
  - Combo box
  - Pop up menu
  - Pull down menu



Mini Controls

- Lucida Grande 9
- NSFont method to support control sizes

```
+ (float)systemFontSizeForControlSize:(NSControlSize)cs;
```

New Control—"That Switcher Thing"

- The view mode selector in Finder (and now Open/Save)
- Subclass of NSControl
  - Also available as a cell

Help name this new class!

Send your name ideas to:

NameThatClass@group.apple.com

New Control—“That Switcher Thing”

- By submitting your name ideas:
  - Your product name submission automatically becomes the property of Apple, without compensation to you
  - You will not challenge or interfere with Apple's use for such name or similar name anywhere in the world
  - You are not using or intend to use your name submission, and are not aware of any use or intent to use such name by any third party

New Button Styles

- Disclosure triangle

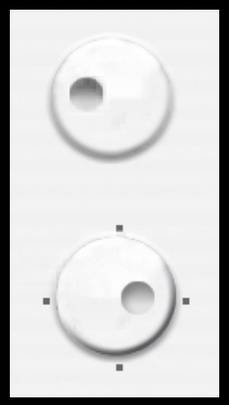
```
NSDisclosureBezelStyle
```

- “Textured” button

```
NSTexturedSquareBezelStyle
```

Circular Slider

```
- (void)setSliderType:(NSSliderType)sliderType;
- (NSSliderType)sliderType;

typedef enum {
  NSLinearSlider,
  NSCircularSlider
} NSSliderType;
```



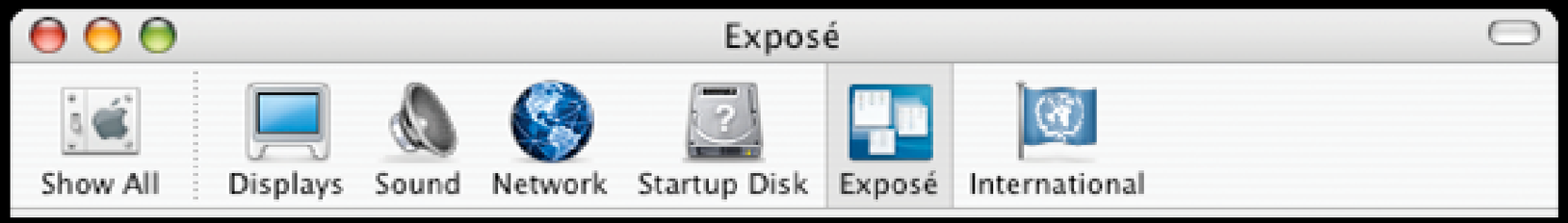
NSToolbar

- Support for “selectable” toolbar items
- Delegate implements

```
- (NSArray *)toolbarSelectableItemIdentifiers:(id)bar;
```

- Can change selected item with

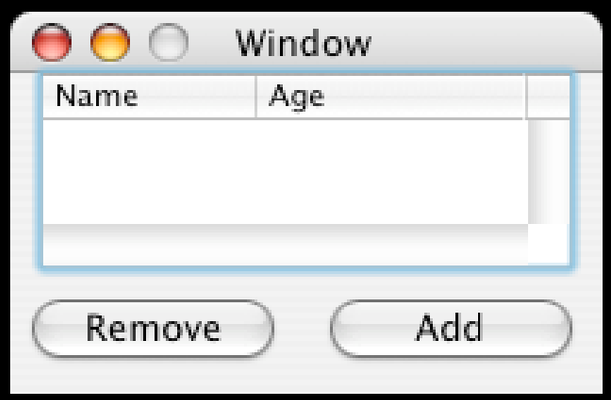
```
- (void)setSelectedItemIdentifier:(NSString *)itemID;
- (NSString *)selectedItemIdentifier;
```



NSCell and NSView

- Controlling focus ring drawing

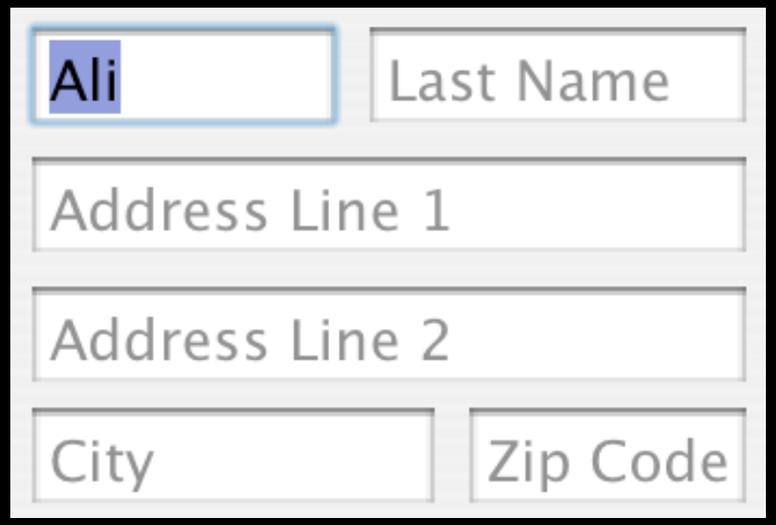
```
-(void)setFocusRingType:(NSFocusRingType)type;

typedef enum {
    NSFocusRingTypeDefault,
    NSFocusRingTypeNone,
    NSFocusRingTypeExterior
} NSFocusRingType;
```



NSTextFieldCell

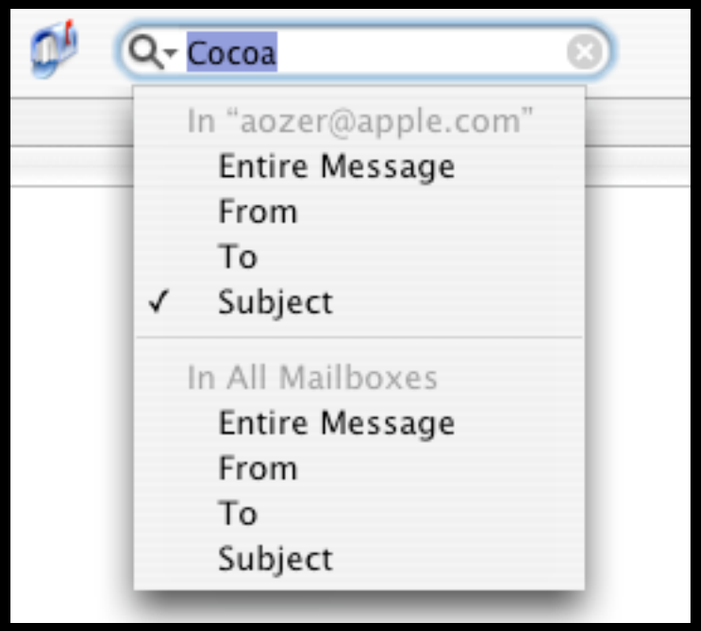
- API for “placeholder” string
  - String to be displayed when empty and not editing

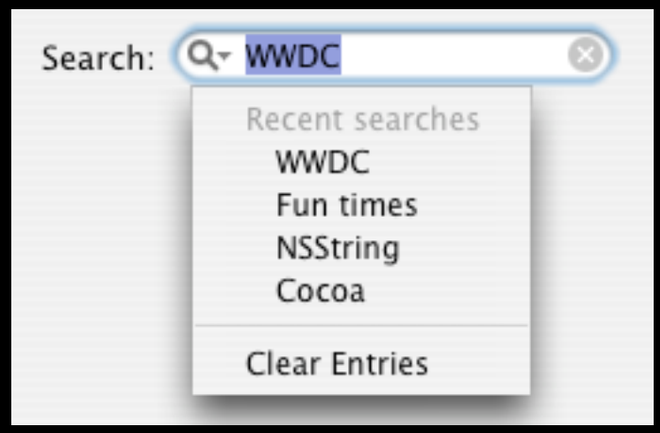
```
- (void)setPlaceholderString:(NSString *)string;
- (NSString *)placeholderString;
```



NSSearchField

- New control
  - Subclass of NSTextField
  - Also NSSearchFieldCell
-
- Features
  - Search button with menu
  - Cancel button
  - Recents list



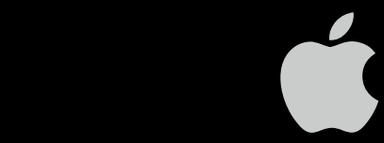


NSSearchField

- Customize the search or cancel buttons

```
- (void)setSearchButtonCell:(NSButtonCell *)cell;
- (NSButtonCell *)searchButtonCell;

- (void)setCancelButtonCell:(NSButtonCell *)cell;
- (NSButtonCell *)cancelButtonCell;
```

```
- (void)setSearchMenuTemplate:(NSMenu *)menu;
- (NSMenu *)searchMenuTemplate;
```



NSSearchField

- Customize recents

```
- (void)setRecentSearches:(NSArray*)searches;
- (NSArray*)recentSearches;

- (void)setRecentsAutosaveName:(NSString*)name;
- (NSString*)recentsAutosaveName;

- (void)setMaximumRecents:(int)maxRecents;
- (int)maximumRecents;
```

NSFontPanel

- Many improvements from a user point of view
  - Improved collection management
  - Better user interface
  - Color support in favorites
  - Ability to search fonts

NSFontPanel

- Ability to enable/disable elements of the UI:

```
-(unsigned)validModesForFontPanel:(NSFontPanel *)panel;

enum {
    NSFontPanelFaceModeMask,
    NSFontPanelSizeModeMask,
    NSFontPanelCollectionModeMask,
    NSFontPanelStandardModesMask,
    NSFontPanelAllModesMask
};
```

NSFontManager

- APIs for managing collections

```
- (BOOL)addCollection:(NSString *)name options:(int)opts;

- (BOOL)removeCollection:(NSString *)name;

- (NSArray *)collectionNames;
```

NSFontManager

- APIs for managing fonts in collections

```
- (void)addFontDescriptors:(NSArray *)descArray
              toCollection:(NSString *)name;

- (void)removeFontDescriptor:(NSFontDescriptor *)desc
              fromCollection:(NSString *)name;

- (NSArray *)fontDescriptorsInCollection:(NSString *)c;
```

NSFontDescriptor

- Describes a font

```
- (id)initWithFontAttributes:(NSDictionary *)attributes;

NSString *NSFontFamilyAttribute;
NSString *NSFontNameAttribute;
NSString *NSFontFaceAttribute;
NSString *NSFontSizeAttribute;
NSString *NSFontVisibleNameAttribute;
NSString *NSFontColorAttribute;
```

Text

- Lots of changes in text
- Session 427 “Cocoa Text” will cover these in detail

Text Demo

Text

- Many new user features!
  - Read/write support for Word documents
  - Standard find panel
  - Enumerate document styles
  - Save favorite styles
  - Better text ruler
  - Background color
  - Text completion
  - Setting base writing direction
  - Character styles: Outline, shadow, …

Text

- New text attributes
  - Control over stroke, color, width
  - Strikethrough and underline colors
  - Shadow
  - Obliqueness
  - Expansion
  - Cursor
  - Tool tip
  - Line heights
  - Paragraph spacing

Text

- New Classes
  - NSATSTypesetter
  - NSGlyphGenerator
- Additional Changes
  - Truncated text drawing
  - Improved accessibility
  - Performance improvements
- Session 427!

NSShadow

- New class for specifying a drop shadow when drawing
- Always drawn in base space
  - Rotations, translations don’t affect the drawn shadow
- Value for new text attribute

```
NSShadowAttributeName
```

NSShadow

- Setting offset

```
-(void)setShadowOffset:(NSSize)offset;
-(NSSize)shadowOffset;
```

- Setting blur

```
-(void)setShadowBlurRadius:(float)val;
-(float)shadowBlurRadius;
```

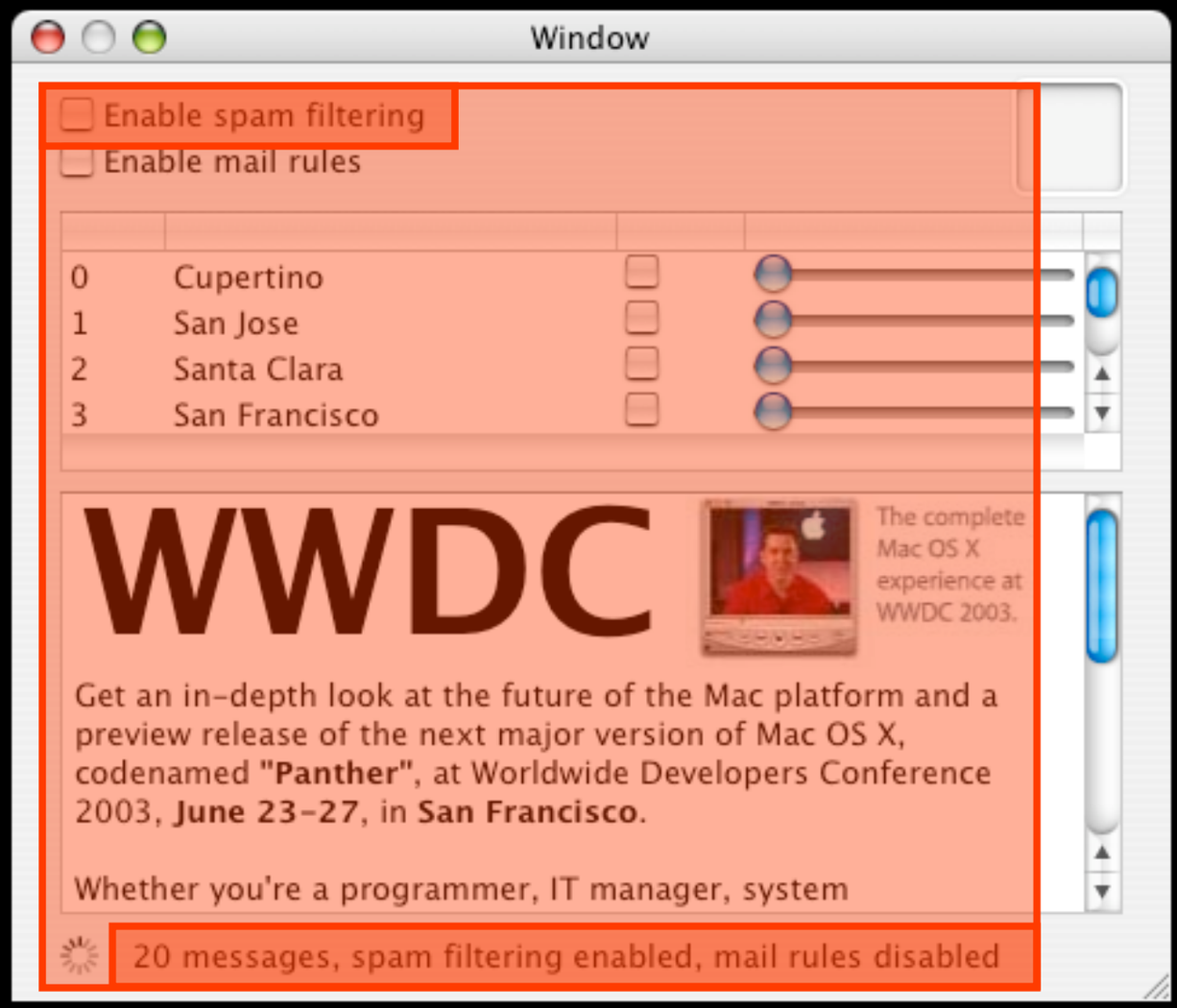
- Using a shadow

```
-(void)set;
```

NSView

- Jaguar:
  - NSView maintains single “dirty” rectangle
  - May redraw more than is strictly necessary

NSView


Window
Enable spam filtering
Enable mail rules
0 Cupertino
1 San Jose
2 Santa Clara
3 San Francisco
WWDC
The complete Mac OS X experience at WWDC 2003.
Get an in-depth look at the future of the Mac platform and a preview release of the next major version of Mac OS X, codenamed "Panther", at Worldwide Developers Conference 2003, June 23–27, in San Francisco.
Whether you're a programmer, IT manager, system
20 messages, spam filtering enabled, mail rules disabled

NSView

- Panther:
  - Views track invalidated areas
  - Clipping restricted to area that needs drawing only
  - drawRect: is still the funnel point for drawing!

NSView

- Getting the full list of rectangles to draw:

```
-(void)getRectsBeingDrawn:(const NSRect **)rects
                    count:(int *)count;
```

- Does a rect need to be drawn?

```
-(BOOL)needsToDrawRect:(NSRect)aRect;
```

- Turning custom clipping:

```
-(BOOL)wantsDefaultClipping;
```

NSView

- View hiding
  - Individual views
  - Entire view subtrees
  - Cursor rects, tooltip rects, and tracking rects are disabled
  - Responder chain is adjusted automatically

NSView

- Hiding views

```
- (void)setHidden:(BOOL)flag;
```

- Checking to see if a view is hidden

```
- (BOOL)isHidden;
- (BOOL)isHiddenOrHasHiddenAncestor;
```

NSView

- More NSView and graphics in Session 413, “Cocoa 2D Drawing Techniques”

NSWindow

- Custom sheet location

```
-(NSRect)window:(NSWindow *)window
          willPositionSheet:(NSWindow *)sheet
          usingRect:(NSRect)rect;
```

- Custom minimized doc icon image

```
- (void)setMiniwindowImage:(NSImage *)image;
- (NSImage *)miniwindowImage;
```

NSWindow

- Enable tool tips in a window even in inactive apps

```
- (void)setAllowsToolTipsWhenApplicationIsInactive:
                                                (BOOL)flag;

- (BOOL)allowsToolTipsWhenApplicationIsInactive;
```

NSWindow

- Compute a window's frame or content size

```
-(NSRect)frameRectForContentRect:(NSRect)contentRect;
-(NSRect)contentRectForFrameRect:(NSRect)frameRect;
```

NSWindow

- New content-based size methods:

```
-(void)setContentMaxSize:(NSSize)size;
-(NSSize)contentMaxSize;

-(void)setContentMinSize:(NSSize)size;
-(NSSize)contentMinSize;

-(void)setContentAspectRatio:(NSSize)ratio;
-(NSSize)contentAspectRatio;

-(void)setContentResizeIncrements:(NSSize)increments;
-(NSSize)contentResizeIncrements;
```

NSAlert

- New public class
- Adds flexibility over the existing C-based API
  - Assign app-specific key equivalents and return values to the alert's buttons
  - Show a help button
  - Provide more than three buttons (if you really need it…)

NSAlert

- Create one with init, and customize with methods like:

```
- (void)setMessageText:(NSString *)messageText;
- (void)setInformativeText:(NSString *)informativeText;
- (void)setIcon:(NSImage *)icon;
- (void)setShowsHelp:(BOOL)showsHelp;
- (NSButton *)addButtonWithTitle:(NSString *)title;
```

```
-(int)runModal;

-(void)beginSheetModalForWindow:(NSWindow *)window
            modalDelegate:(id)delegate
            didEndSelector:(SEL)didEndSelector
            contextInfo:(void *)contextInfo;
```

NSAlert

- Creation API for easier migration

```
+(NSAlert *)alertWithMessageText:(NSString *)message
         defaultButton:(NSString *)defaultButton
         alternateButton:(NSString *)altButton
         otherButton:(NSString *)otherButton
         informativeTextWithFormat:(NSString *)fmt, ...;
```

NSAlert

- Help Button
  - Provide help anchor
  -

```
- (void)setHelpAnchor:(NSString *)anchor;
- (NSString *)helpAnchor;
```

  - Or more sophisticated handling via delegate method

```
- (BOOL)alertShowHelp:(NSAlert *)alert;
```

NSHelpManager

- More control over accessing help
- Open a specific help book location
-

```
- (void)openHelpAnchor:(NSString *)anchor
                inBook:(NSString *)book;
```

Search for a query string in a book

```
- (void)findString:(NSString *)query
            inBook:(NSString *)book;
```

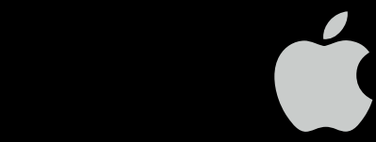
NSWorkspace

- Launching by bundle ID:

```
- (BOOL)launchAppWithBundleIdentifier:(NSString *)bundleID
   options:(NSWorkspaceLaunchOptions)options
   additionalEventParamDescriptor:(NSAppleEventDescriptor *)d
   launchIdentifier:(NSNumber **)identifier;
```

- Opening multiple files:

```
- (BOOL)openURLs:(NSArray *)urls
   withAppBundleIdentifier:(NSString *)bundleID
   options:(NSWorkspaceLaunchOptions)options
   additionalEventParamDescriptor:(NSAppleEventDescriptor *)d
   launchIdentifiers:(NSArray **)identifiers;
```



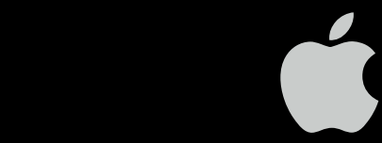
NSWorkspace

- Discovering an application with a bundle ID:

```
- (NSString *)absolutePathForAppBundleWithIdentifier:
                                        (NSString *)bundleID;
```

- NSWorkspaceDidLaunchApplicationNotification dictionary has a new key:

```
NSApplicationBundleIdentifier
```



NSWorkspace

- Sleep notifications

```
NSWorkspaceWillSleepNotification
NSWorkspaceDidWakeNotification
```

NSApplication

- Opening and printing multiple files at once

```
-(void)application:(NSApplication *)app
         openFiles:(NSArray *)files;

-(void)application:(NSApplication *)app
        printFiles:(NSArray *)files;
```

- Delegate handles UI and error conditions

NSApplication

- Replying to the AppleEvent for opening and printing

```
-(void)replyToOpenOrPrint:(NSApplicationDelegateReply)r;

typedef enum {
  NSApplicationDelegateReplySuccess,
  NSApplicationDelegateReplyCancel,
  NSApplicationDelegateReplyFailure
} NSApplicationDelegateReply;
```

NSMenu

- Selectable non-leaf menu items
  - Just set target and action on the NSMenuItem
-
- Dynamic menu items (based on key equivalent)

```
-(void)setAlternate:(BOOL)flag;
-(BOOL)isAlternate;
```

NSMenu

- Attributed menu titles

```
-(void)setAttributedTitle:(NSAttributedString *)attrStr;
-(NSAttributedString *)attributedTitle;
```

- Tool tips

```
-(void)setToolTip:(NSToolTip *)str;
-(NSString *)toolTip;
```

NSMenu

- NSMenu now has a delegate!
-
- For lazily populating the menu:

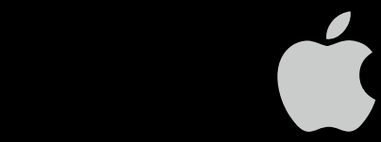
```
- (void)menuNeedsUpdate:(NSMenu *)menu;
```

NSMenu

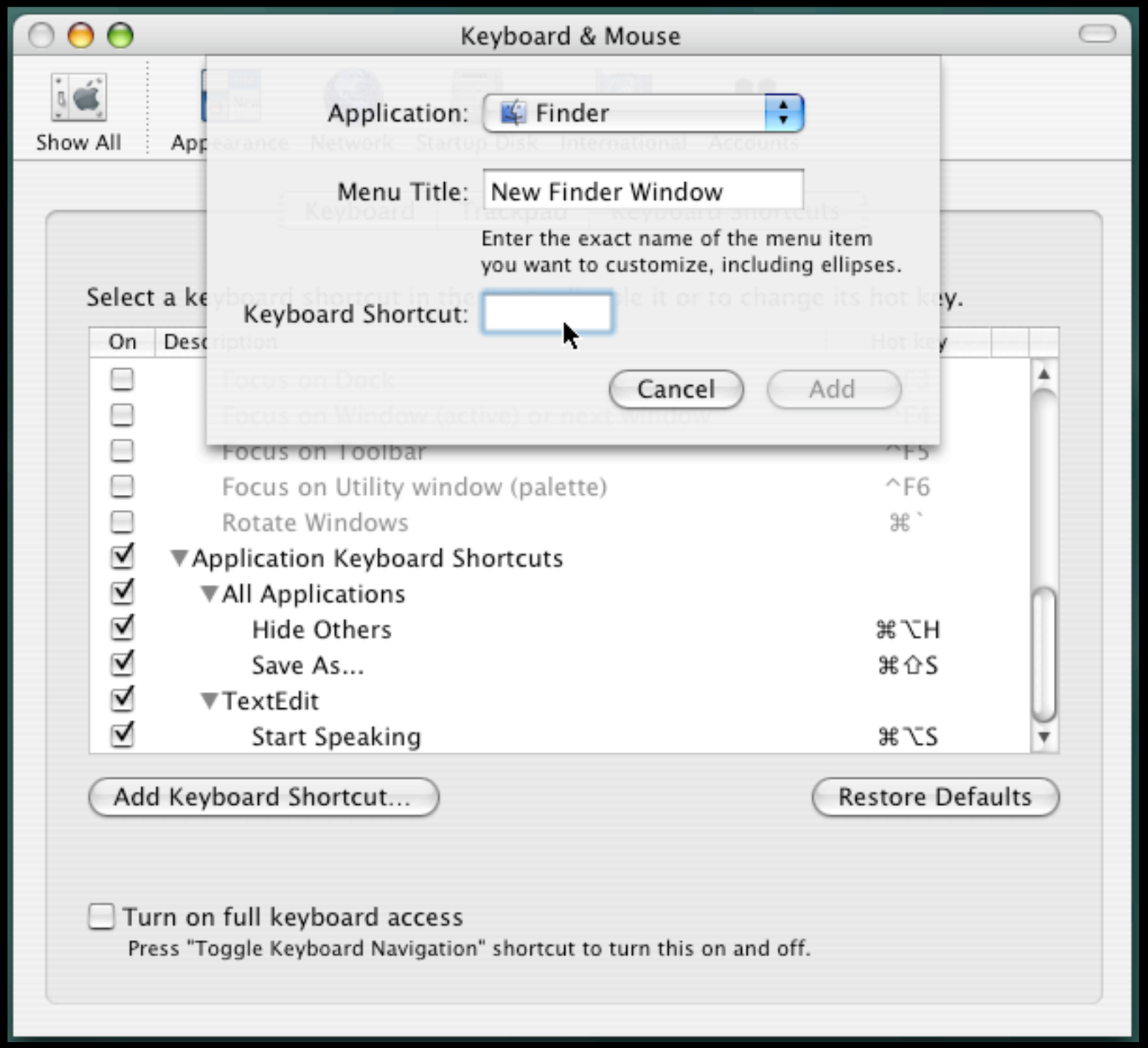
- If building the menu might take a while:

```
- (int)numberOfItemsInMenu:(NSMenu *)menu;
- (BOOL)menu:(NSMenu *)menu
        updateItem:(NSMenuItem *)item
        atIndex:(int)index
        shouldCancel:(BOOL)shouldCancel;
```

```
- (BOOL)menuHasKeyEquivalent:(NSMenu *)menu
        forEvent:(NSEvent *)item
        target:(id *)target
        action:(SEL *)action;
```



Customizable Command Keys



NSBrowser

- Continuous horizontal scrolling
  - On by default
-
- Column resizing

```
-(void)setColumnResizingType:
                    (NSBrowserColumnResizingType)resize;
-(NSBrowserColumnResizingType)columnResizingType;

typedef enum {
    NSBrowserNoColumnResizing,
    NSBrowserAutoColumnResizing,
    NSBrowserUserColumnResizing
} NSBrowserColumnResizingType;
```

NSBrowser

- Column width persistence

```
- (void)setColumnsAutosaveName:(NSString *)name;
- (NSString *)columnsAutosaveName;
+ (void)removeSavedColumnsWithAutosaveName:(NSString *)n;
```

- More complex persistence is possible

```
-(void)browserColumnConfigurationDidChange:
                                (NSNotification *)notification;
```

NSTableView

- NSIndexSet-based selection management

```
- (void)selectColumnIndexes:(NSIndexSet *)indexes
      byExtendingSelection:(BOOL)extend;
- (void)selectRowIndexes:(NSIndexSet *)indexes
      byExtendingSelection:(BOOL)extend;
```

```
-(NSIndexSet *)selectedColumnIndexes;
-(NSIndexSet *)selectedRowIndexes;
```

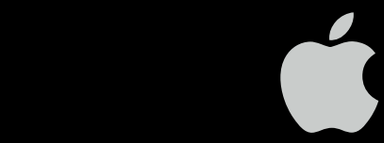
NSTableView

- API for managing sorting UI
- Columns are sortable if they have a descriptor prototype
-

```
-(void)setSortDescriptorPrototype:(NSSortDescriptor *)d;
-(NSSortDescriptor *)sortDescriptorPrototype;
```

- Array of sort descriptors specify the current state
-

```
-(void)setSortDescriptors:(NSArray *)array;
-(NSArray *)sortDescriptors;
```



NSTableView

- Data source actually does the sorting:

```
-(void)tableView:(NSTableView *)tableView
      sortDescriptorsDidChange:(NSArray *)oldDescs;
```

- Outline view, too!

```
-(void)outlineView:(NSOutlineView *)outlineView
      sortDescriptorsDidChange:(NSArray *)oldDescs;
```

NSTableView

- Finer grid drawing control
-

```
- (void)setGridStyleMask:(unsigned)mask;
```

- Control over background drawing
-

```
- (void)drawBackgroundInClipRect:(NSRect)rect;
```

- Drawing background using standard alternating colors

```
- (void)setUsesAlternatingRowBackgroundColors:(BOOL)flag;
- (BOOL)usesAlternatingRowBackgroundColors;
```

NSColor

- Standard colors for alternating rows in lists

```
+ (NSArray *)controlAlternatingRowBackgroundColors;
```

- Finding out the current Aqua tint

```
+ (NSControlTint)currentControlTint;
```

- Ability to specify fill and stroke colors independently

```
- (void)setFill;
- (void)setStroke;
```

NSImage

- Now supports
  - PostScript and EPS files
  - Five channel CMYKA images

NSImageView

- Automatic playback of animated images

```
- (void)setAnimates:(BOOL)flag;
- (BOOL)animates;
```

- Timing and looping specified by image data

Poof Animation

- Poof it like the pros

```
void NSShowAnimationEffect(
        NSAnimationEffect effect,
        NSPoint centerLocation,
        NSSize size,
        id animationDelegate,
        SEL didEndSelector,
        void *contextInfo);

typedef enum {
    NSAnimationEffectDisappearingItemDefault,
    NSAnimationEffectPoof
} NSAnimationEffect;
```

NSCursor

- Now supports cursor images other than 16x16
- Provides additional standard cursors

```
+ (NSCursor *)pointingHandCursor;
+ (NSCursor *)closedHandCursor;
+ (NSCursor *)openHandCursor;
+ (NSCursor *)resizeLeftCursor;
+ (NSCursor *)resizeRightCursor;
+ (NSCursor *)resizeLeftRightCursor;
+ (NSCursor *)resizeUpCursor;
+ (NSCursor *)resizeDownCursor;
+ (NSCursor *)resizeUpDownCursor;
+ (NSCursor *)crosshairCursor;
```

Speech

- Two new classes for handling speech
  - NSSpeechRecognizer
  - NSSpeechSynthesizer

Controller Layer (“Bindings”)

- New APIs to eliminate controller “glue” code
  - Enhancements to key value coding (KVC)
  - New class NSController and subclasses
    - NSObjectController
    - NSArrayController
    - NSUserDefaultsController
  - Tight integration with Interface Builder

Controller Layer (”Bindings”)

- Session 412, “Cocoa UI Programming”
- Session 419, “Cocoa UI Programming in Depth”
  – 419 is a recent addition—Friday 9am

Interface Builder

- Support for many of the features/APIs discussed here
- Bindings
- WebKit
- Improved compatibility checking

Interface Builder Demo

NSNib

- New class representing an Interface Builder “nib” file
- Load a nib file once and instantiate it many times

```
- (id)initWithContentsOfURL:(NSURL *)nibFileURL;

- (BOOL)instantiateNibWithOwner:(id)owner
                 topLevelObjects:(NSArray **)objects;
```

- Can also archive/unarchive a nib file

Foundation Update

Chris Parker
Engineer
Cocoa Frameworks

New Exceptions

- Java style, including finally clauses
- Binary compatible
- Automatically
  - promotes variables to “volatile”
  - deals with return statements
  - deals with continue/break/goto statements

New Exceptions

```
NS_DURING
    ...
    NS_VALUERETURN(object, id)
NS_HANDLER
    ...
NS_ENDHANDLER
```

New Exceptions

```
NS_DURING
    ...
    NS_VALUERETURN(object, id)
NS_HANDLER
    ...
NS_ENDHANDLER
```

```
@try {
    ...
    return object;
}
@catch (NSException *e) {
    ...
}
```

Synchronization

- New statement
- Synchronize on any object
- Exceptions clear locks held above catch point

```
@synchronized(object) {
    ...
}
```

Key-Value Coding

- Some older API has been deprecated
- New API to support bindings
- New, more consistently named methods

```
-(void)setValue:(id)value forKey:(NSString *)key;

-(BOOL)validateValue:(id *)ioValue
        forKey:(NSString *)inKey
        error:(NSError **)outError;

-(NSMutableArray *)mutableArrayValueForKey:(NSString *)key;
```

Key-Value Coding

- New API for undefined keys, nil values

```
-(id)valueForUndefinedKey:(NSString *)key;

-(void)setValue:(id)value
      forUndefinedKey:(NSString *)key;

-(void)setNilValueForKey:(NSString *)key;
```

Key-Value Coding

- Categories on NSArray, NSDictionary

```
-(id)valueForKey:(NSString *)key;
```

- Also on NSMutableArray, NSMutableDictionary

```
-(void)setValue:(id)value forKey:(NSString *)key;
```

- Detailed comments

```
NSKeyValueCoding.h
```

Key-Value Observing

- Objects are registered for notification about value changes for keypaths
- For NSObject and subclasses:

```
-(void)addObserver:(NSObject *)observer
        forKeyPath:(NSString *)keyPath
           options:(NSKeyValueObservingOptions)options
           context:(void *)context;
```

```
-(void)addObserver:(NSObject *)observer
        toObjectsAtIndexes:(NSIndexSet *)indexes
        forKeyPath:(NSString *)keyPath
           options:(NSKeyValueObservingOptions)options
           context:(void *)context;
```

Key-Value Observing

- Observer objects are informed of changes to values for a key path:

```
-(void)observeValueForKeyPath:(NSString *)keyPath
      ofObject:(NSObject *)object
      change:(NSDictionary *)change
      context:(void *)context;
```

- Observer notifications can be automatic or manual

NSIndexSet

- Memory-efficient and fast storage of large numbers of unsigned integers

```
-(void)initWithIndex:(unsigned int)value;
-(void)initWithIndexesInRange:(NSRange)range;

-(BOOL)containsIndex:(unsigned)value;
-(BOOL)containsIndexesInRange:(NSRange)range;
-(BOOL)containsIndexes:(NSIndexSet *)indexes;
-(BOOL)intersectsIndexesInRange:(NSRange)range;
```

```
-(unsigned)firstIndex;
-(unsigned)indexGreaterThanIndex:(unsigned int)value;
-(unsigned)indexLessThanOrEqualToIndex:(unsigned)value;
```

NSMutableIndexSet

- Mutable version allows manipulation by individual index

```
-(void)addIndex:(unsigned int)value;
-(void)removeIndex:(unsigned int)value;
```

- By range

```
-(void)addIndexesInRange:(NSRange)range;
-(void)removeIndexesInRange:(NSRange)range;
```

```
-(void)addIndexes:(NSIndexSet *)indexSet;
-(void)removeIndexes:(NSIndexSet *)indexSet;
```

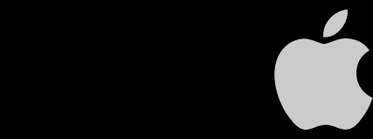
NSSortDescriptor

- Array of NSSortDescriptors describe a sort ordering
- Can be used for comparing objects

```
-(NSComparisonResult)compareObject:(id)object1
                          toObject:(id)object2;
```

```
-(NSArray *)sortedArrayUsingDescriptors:(NSArray *)desc;
```

```
-(void)sortUsingDescriptors:(NSArray *)sortDescriptors;
```



NSValueTransformer

- Value transformers return a new object for a given value
- May return the same or different type

```
-(id)transformedValue:(id)value;
-(id)reverseTransformedValue:(id)value;
```

- Some common transformers are provided

```
NSNegateBooleanTransformerName
NSIsNilTransformerName
NSIsNotNilTransformerName
NSUnarchiveFromDataTransformerName
```

NSError

- New public class for describing errors flexibly
- Allows attaching of userInfo dictionary
- Creating

```
-(id)initWithDomain:(NSString *)domain
     code:(int)code
     userInfo:(NSDictionary *)dict;
```

- Accessors for domain, code, userInfo, localizedDescription

NSXMLParser

- Event-based XML parsing

```
-(id)initWithContentsOfURL:(NSURL *)url;
-(id)initWithData:(NSData *)data;
```

- Delegate implements events it's interested in

```
-(void)parserDidStartDocument:(NSXMLParser *)parser;

-(void)parser:(NSXMLParser *)parser
        didStartElement:(NSString *)elementName
        namespaceURL:(NSString *)namespaceURI
        qualifiedName:(NSString *)qName;

-(void)parser:(NSXMLParser *)parser
        foundComment:(NSString *)comment;
```

NSStream

- NSInputStream and NSOutputStream
- Toll-free bridged to CFReadStream and CFWriteStream
- Supports SSL, proxies, etc.
- File, memory, network streams
- Subclassable
  - Only subclasses of NSInputStream and NSOutputStream are bridged

NSInputStream

- Creating NSInputStreams

```
-(id)initWithData:(NSData *)data;
-(id)initWithFileAtPath:(NSString *)path;
```

- Reading

```
-(int)read:(uint8_t *)buf maxLength:(unsigned)len;

-(BOOL)getBuffer:(uint8_t **)buf length:(unsigned *)len;

-(BOOL)hasBytesAvailable;
```

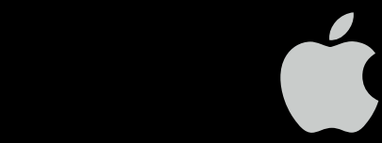
NSOutputStream

- Creating NSOutputStreams

```
-(id)initToMemory;
-(id)initToBuffer:(uint8_t *)buffer
     capacity:(unsigned int)capacity;
-(id)initToFileAtPath:(NSString *)path;
```

- Writing to the stream

```
-(int)write:(const uint8_t *)buffer
      maxLength:(unsigned int)len;

-(BOOL)hasSpaceAvailable;
```



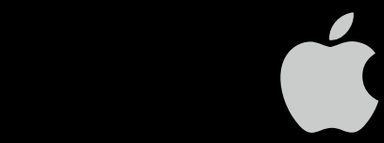
NSStreams to Hosts

- Network streams are essentially pairs

```
+(void)getStreamsToHost:(NSHost *)host
        port:(int)port
        inputStream:(NSInputStream **)inputStream
        outputStream:(NSOutputStream **)outputStream;
```

- Scheduling streams

```
-(void)scheduleInRunLoop:(NSRunLoop *)aRunLoop
        forMode:(NSString *)mode;

-(void)removeFromRunLoop:(NSRunLoop *)aRunLoop
        forMode:(NSString *)mode;
```



NSStream

- Asynchronous state events for scheduled streams are delivered to the delegate

```
-(void)stream:(NSStream *)aStream
      handleEvent:(NSStreamEvent)eventCode;
```

- Event codes

```
NSStreamEventOpenCompleted
NSStreamEventHasBytesAvailable
NSStreamEventHasSpaceAvailable
NSStreamEventErrorOccurred
NSStreamEventEndEncountered
```

NSStream Properties

- Accessing properties

```
-(id)propertyForKey:(NSString *)key;
-(BOOL)setProperty:(id)property forKey:(NSString *)key;
```

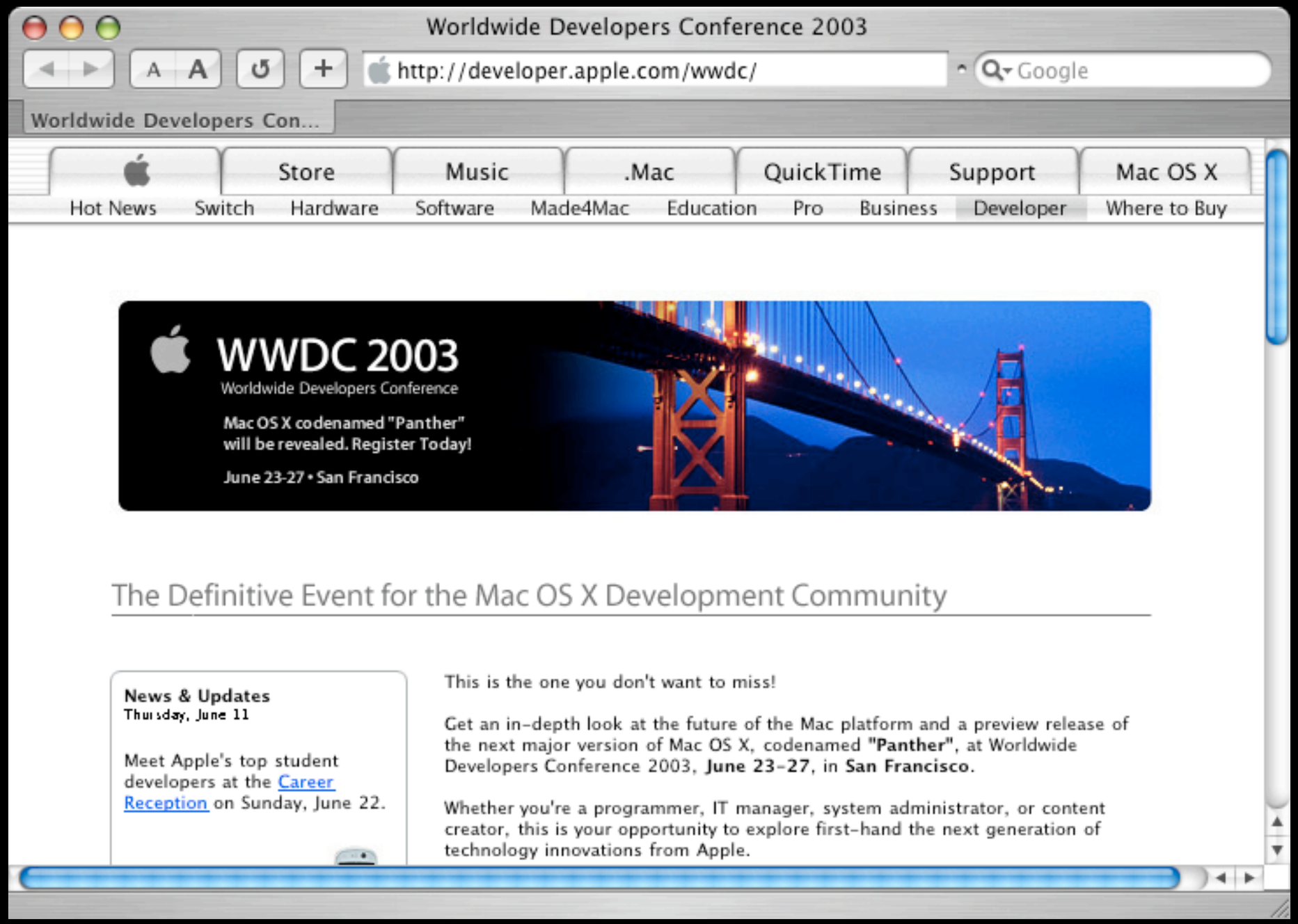
- Properties

```
NSStreamSocketSecurityLevelKey
NSStreamSOCKSProxyHostKey
NSStreamDataWrittenToMemoryStreamKey
NSStreamFileCurrentOffsetKey
```

NSURL Enhancements

- Many new classes
  - NSURL
  - NSURLRequest, NSURLResponse
  - NSURLCache
  - NSURLConnection
  - NSURLDownload
  - NSURLProtocol

Technology Demonstration



Wrap Up

Roadmap

412 Cocoa UI Programming Controller Layer Overview	**Presidio** **Wed., 3:30pm**
413 Cocoa 2D Drawing Techniques NSView and Cocoa Graphics	**Nob Hill** **Wed., 5:00pm**
419 Cocoa Performance Techniques More Controllers	**Russian Hill** **Fri., 9:00am**
424 Cocoa: Tips and Tricks Adding nifty features to your apps	**Mission** **Fri., 2:00pm**
427 Cocoa Text Overview of Cocoa Text and New Features	**Presidio** **Fri., 5:00pm**

Roadmap

400 Introduction to Developing Applications with Cocoa	Nob Hill Mon., 5:00pm
FF008 Cocoa Feedback Forum	North Beach Thu., 10:30am

Who to Contact

John Geleynse
Manager, Software Evangelism
geleynse@apple.com

Cocoa Feedback
For suggestions and feedback
cocoa-feedback@group.apple.com

Developer Technical Support
Code-level questions (fee-based)
dts@apple.com

Help us name “That Switcher Thing”
Email us your name ideas for new class
NameThatClass@group.apple.com

http://developer.apple.com/wwdc2003/urls.html

For More Information

Reference Library

Documentation

Release notes for Foundation and Application Kit

ADC Home > Documentation > Cocoa > Release Notes >

Documentation of new API available in Panther

Q&A

Ali Ozer
Manager
Cocoa Frameworks

http://developer.apple.com/wwdc2003/urls.html

WWDC 2003

Worldwide Developers Conference